



LATINO**WARE** 2022

19º Congresso Latino-americano de
Software Livre e Tecnologias Abertas

02 a 04 de novembro de 2022

Hotel Golden Park Internacional Foz do Iguaçu
Foz do Iguaçu | Paraná | Brasil

Realização:



Crie um teclado(Macropad) para automatizar suas tarefas

Autores:

Fábio Souza

Embarcados

O EMBARCADOS

COMPARTILHAR O
CONHECIMENTO É
**APRENDER TODOS
OS DIAS**

O Embarcados é uma plataforma que contribui para o desenvolvimento tecnológico no Brasil, a partir do compartilhamento de informações na área de sistemas eletrônicos.

Do estudante ao profissional, dos makers aos curiosos, acreditamos que todos devem ter acesso a informações de qualidade – que auxiliam em projetos eletrônicos e servem para atualização do conhecimento.

www.embarcados.com.br



Embarcados

TRAJETÓRIA • CONTEÚDO, CONCURSOS E AÇÕES

O PORTAL

A ATUALIZAÇÃO DO PORTAL É CONSTANTE,
BEM COMO O SEU ENVOLVIMENTO!

Todos os dias, publicamos artigos com conteúdos bem diversificados na área de sistemas embarcados, que contêm informações relativas a:



Artigos

Sobre sistemas embarcados



Projetos

Hardware e firmware/software



Reviews

De placas, processadores e CIs



Estudos

Sobre Estado da Arte, história e teoria da eletrônica, além de Sistemas Digitais (FPGA, Verilog e VHDL)



Dicas e truques

Que auxiliam em projetos e em partes específicas de projetos



Notícias

Notícias atuais sobre a área de sistemas embarcados



Fórum

A grande comunidade que ensina e aprende através de sistemas embarcados!



www.embarcados.com.br

Franzininho



O projeto Franzininho tem como missão capacitar os brasileiros para se tornarem protagonistas no desenvolvimento de projetos eletrônicos em nosso país.

Disponibilizamos materiais que tornam o estudo e aplicações das tecnologias atuais acessíveis para a sociedade brasileira através de conteúdos, materiais didáticos, ferramentas de desenvolvimento, projetos compartilhados e uma comunidade envolvida.

Acreditamos que projetos open source, materiais didáticos abertos, compartilhamento de conhecimento, experimentações e colaboração podem impactar positivamente em nossa sociedade.

franzininho.com.br

Tópicos

- Por que você precisa de um teclado de atalhos?
- Como fazer um teclado?
- Ideias de teclados

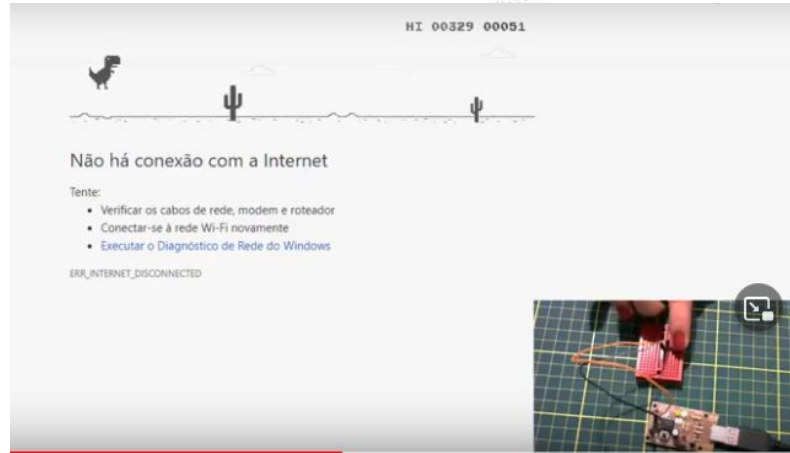
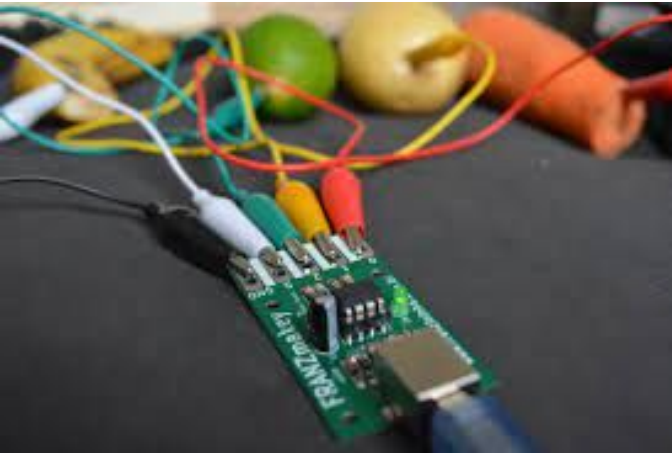
Por que você precisa de um teclado de atalhos?



Por que você precisa de um teclado de atalhos?

- Automação de tarefas (Atalhos, mouse, comandos, etc)
- Jogar
- Testes de comandos em Aplicações
- Interfaces diferentes com PC/Smartphone (Touch, Sensores, etc)
- Adaptação para limitações motoras (PCD, ELA (Esclerose Lateral Amiotrófica))
- Use sua criatividade

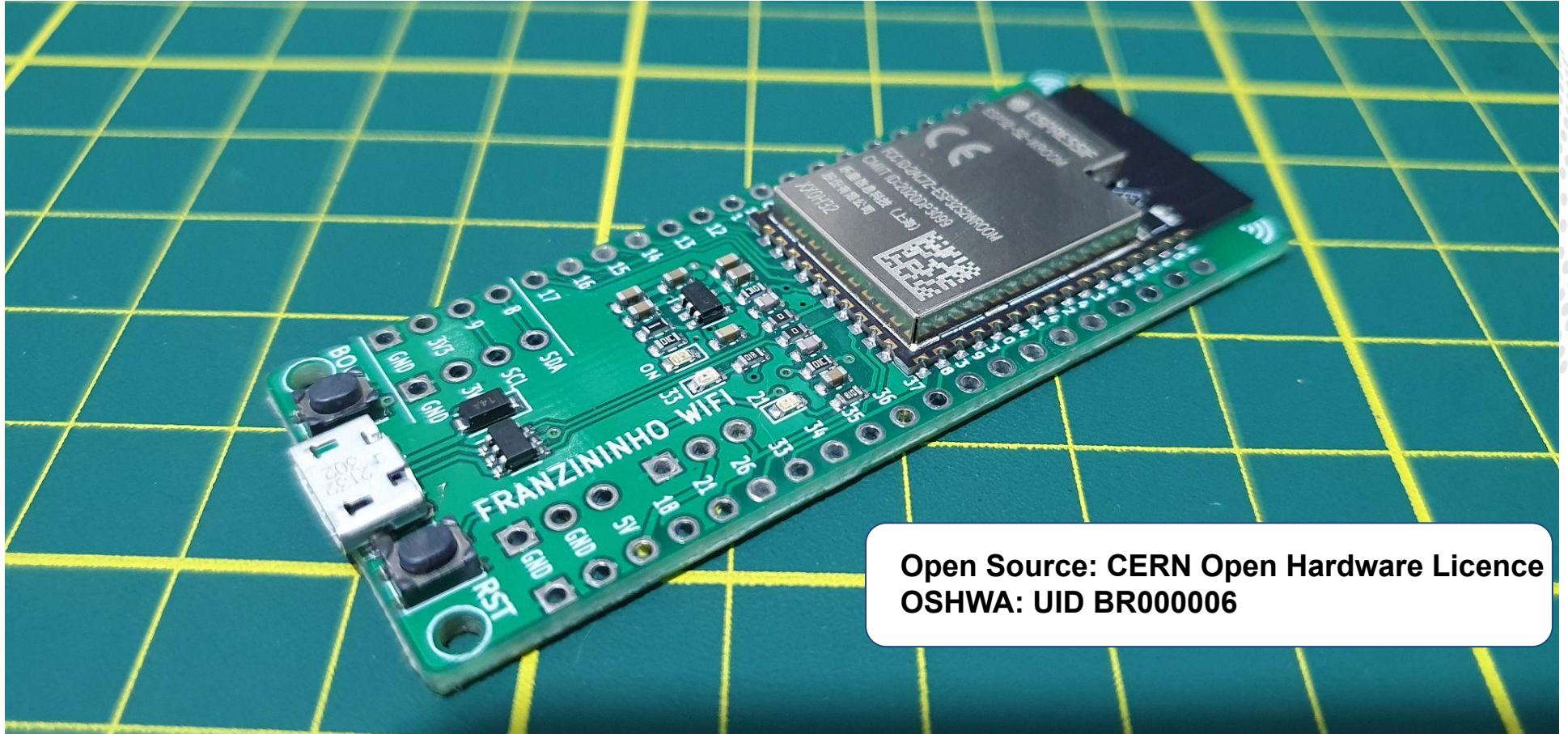
Motivação



O que preciso para fazer um teclado?

- Microcontrolador com USB device com suporte a HID
- Circuito para interface com o usuário
- Biblioteca para interface HID
- Firmware para controle da aplicação

Franzininho WiFi



**Open Source: CERN Open Hardware Licence
OSHWA: UID BR000006**

Franzininho WiFi - Recursos

- Módulo ESP32-S2 – WROOM
 - Microcontrolador Xtensa® single-core32-bit LX7 operando em até 240 MHz
 - Memórias: 128 KB de ROM, 320 KB de SRAM, 16 KB SRAM no RTC, 4MB de Flash
 - WiFi 802.11 b/g/n
 - Interfaces: GPIO, SPI, LCD, UART, I2C, I2S, Camera, IR, contador de pulsos, LED PWM, TWAI(CAN), USB 1.1 OTG, ADC, DAC, touch, sensor de temperatura interno.
- Pinout:
 - 40 pinos divididos em 2×20 headers de 2,54 mm
 - 35 GPIOs
 - Compatível com protoboard
- 2 LEDS de uso geral
- Botões: 1xReset, 1 x Boot
- USB: conector micro USB – OTG 1.1
- Alimentação:
 - 5V via conector USB
 - 5V e GND via pinos
 - 3V3 e GND via pinos
- Programação: ESP-IDF, Arduino, CircuitPython e MicroPython

Franzininho WiFi - Programação

espressif/esp-idf

Espressif IoT Development Framework. Official development framework for ESP32.

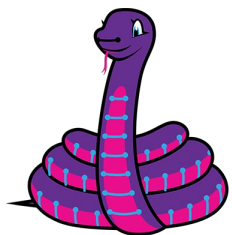


463
Contributors

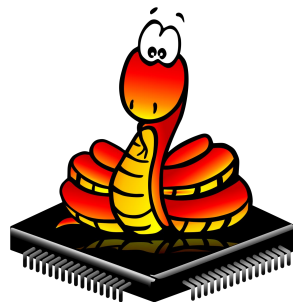
874
Issues

7k
Stars

4k
Forks



circuit
python

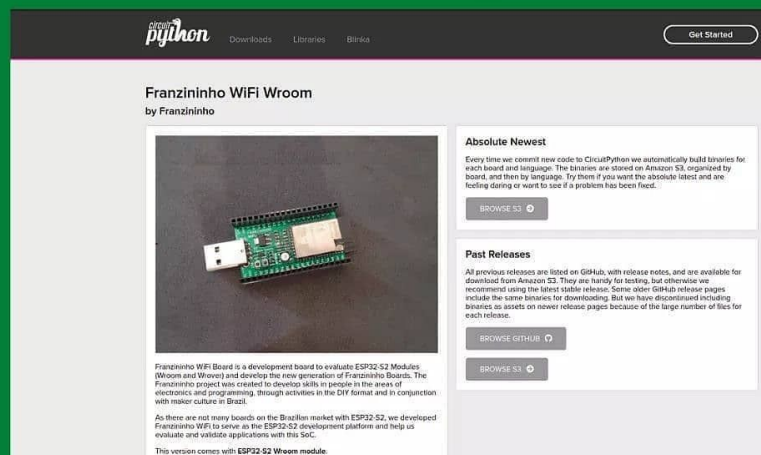


CircuitPython

- O CircuitPython foi projetado tendo em mente a educação;
- Facilidade na edição e manutenção de códigos;
- Serial Console + REPL;
- Armazenamento interno para registro de dados, reprodução de áudio e outras formas de interação com arquivos;
- Mais de 280 bibliotecas;
- Tradução para diversas línguas

CircuitPython


FRANZININHO WIFI NO CIRCUITPYTHON



The screenshot shows the project page for 'Franzininho WiFi Wroom' by Franzinho on the CircuitPython website. The page has a dark header with the CircuitPython logo and navigation links for Downloads, Libraries, and Boards. A 'Get Started' button is in the top right. The main content area features a photo of the ESP32-S2 Wroom module. Below the photo is a description: 'Franzininho WiFi Board is a development board to evaluate ESP32-S2 Modules (Wroom and Wroom) and develop the new generation of Franzinho Boards. The Franzinho project was created to develop skills in people in the areas of electronics and programming, through activities in the DIY format and in conjunction with maker culture in Brazil.' It also mentions that the board is used as a development platform for the ESP32-S2 and comes with an ESP32-S2 Wroom module. To the right of the description are two sections: 'Absolute Newest' with a 'Browse S3' button, and 'Past Releases' with a 'Browse Github' button and a 'Browse S3' button.




CircuitPython

[Downloads](#)[Libraries](#)[Blinka](#)[Get Started](#)

Franzininho WiFi Wroom

by Franzininho



Franzininho WiFi Board is a development board to evaluate ESP32-S2 Modules (Wroom and Wrover) and develop the new generation of Franzininho Boards. The Franzininho project was created to develop skills in people in the areas of electronics and programming, through activities in the DIY format and in conjunction with maker culture in Brazil.

As there are not many boards on the Brazilian market with ESP32-S2, we developed Franzininho WiFi to serve as the ESP32-S2 development platform and help us evaluate and validate applications with this SoC.


CircuitPython 7.0.0


This is the latest **stable** release of CircuitPython that will work with the Franzininho WiFi Wroom.

Start here if you are new to CircuitPython.

[Release Notes for 7.0.0](#)

PORTUGUESE (BRAZILIAN) ▾

DOWNLOAD .BIN NOW 

DOWNLOAD .UF2 NOW 

Built-in modules available: _bleio, adafruit_bus_device, adafruit_pixelbuf, aesio, alarm, analogio, atexit, audiobusio, audiocore, audiomixer, binascii, bitbangio, bitmaptools, board, busio, canio, countio, digitalio, displayio, dualbank, errio, fontio, framebufferio, frequencyio, getpass, imagecapture, ipaddress, json, keypad, math, microcontroller, msgpack, neopixel_write, nvme, onewireio, os, paralleldisplay, ps2io, pulseio, pwmio, qrio, rainbowio, random, re, rgbmatrix, rotaryio, rtc, sdcardio, sharpdisplay, socketpool, ssl, storage, struct, supervisor, synthio, terminalio, time, touchio, traceback, ulab, usb_cdc, usb_hid, usb_midi, vectorio, watchdog, wifi

Absolute Newest

Every time we commit new code to CircuitPython we automatically build binaries for each board and language. The binaries are stored on Amazon S3, organized by board, and then by language. Try them if you want the absolute latest and are feeling daring or want to see if a problem has been fixed.

<https://circuitpython.org/downloads?q=franzininho>

CircuitPython

```
""" Hello, World! (Pisca LED) """
import board
import time
from digitalio import DigitalInOut, Direction

# Configurando o pino do LED, o led que configurei foi o pino 4 = I04
led = DigitalInOut(board.I04)
led.direction = Direction.OUTPUT

#loop infinito - executando sempre
while True:
    led.value = True
    time.sleep(0.5)
    led.value = False
    time.sleep(0.5)
```

Copy

Mais detalhes em: <https://docs.franzininho.com.br>

CircuitPython

```
""" Hello, World! (Pisca LED) """
import board
import time
from digitalio import DigitalInOut, Direction

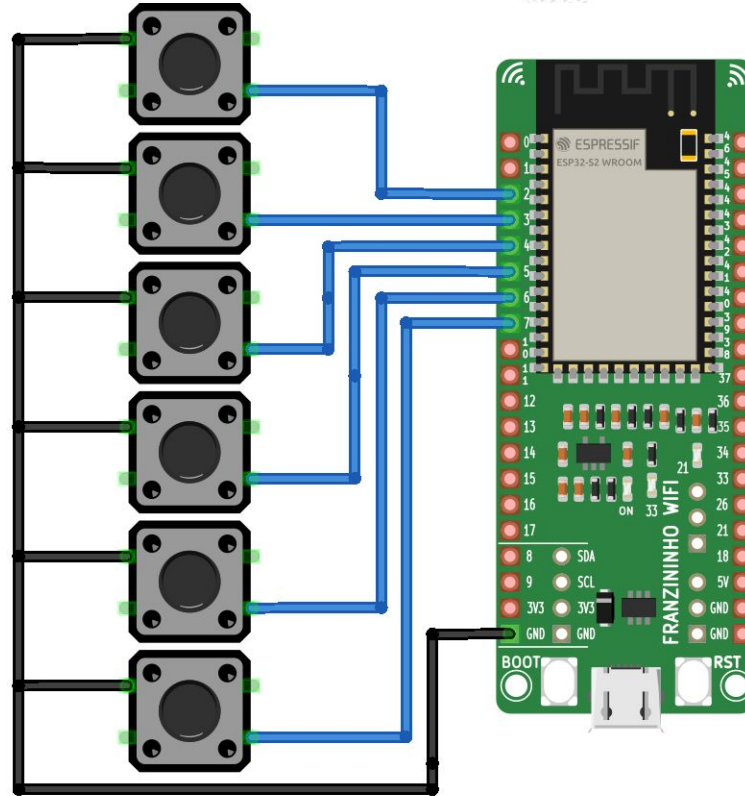
# Configurando o pino do LED, o led que configurei foi o pino 4 = I04
led = DigitalInOut(board.I04)
led.direction = Direction.OUTPUT

#loop infinito - executando sempre
while True:
    led.value = True
    time.sleep(0.5)
    led.value = False
    time.sleep(0.5)
```

Copy

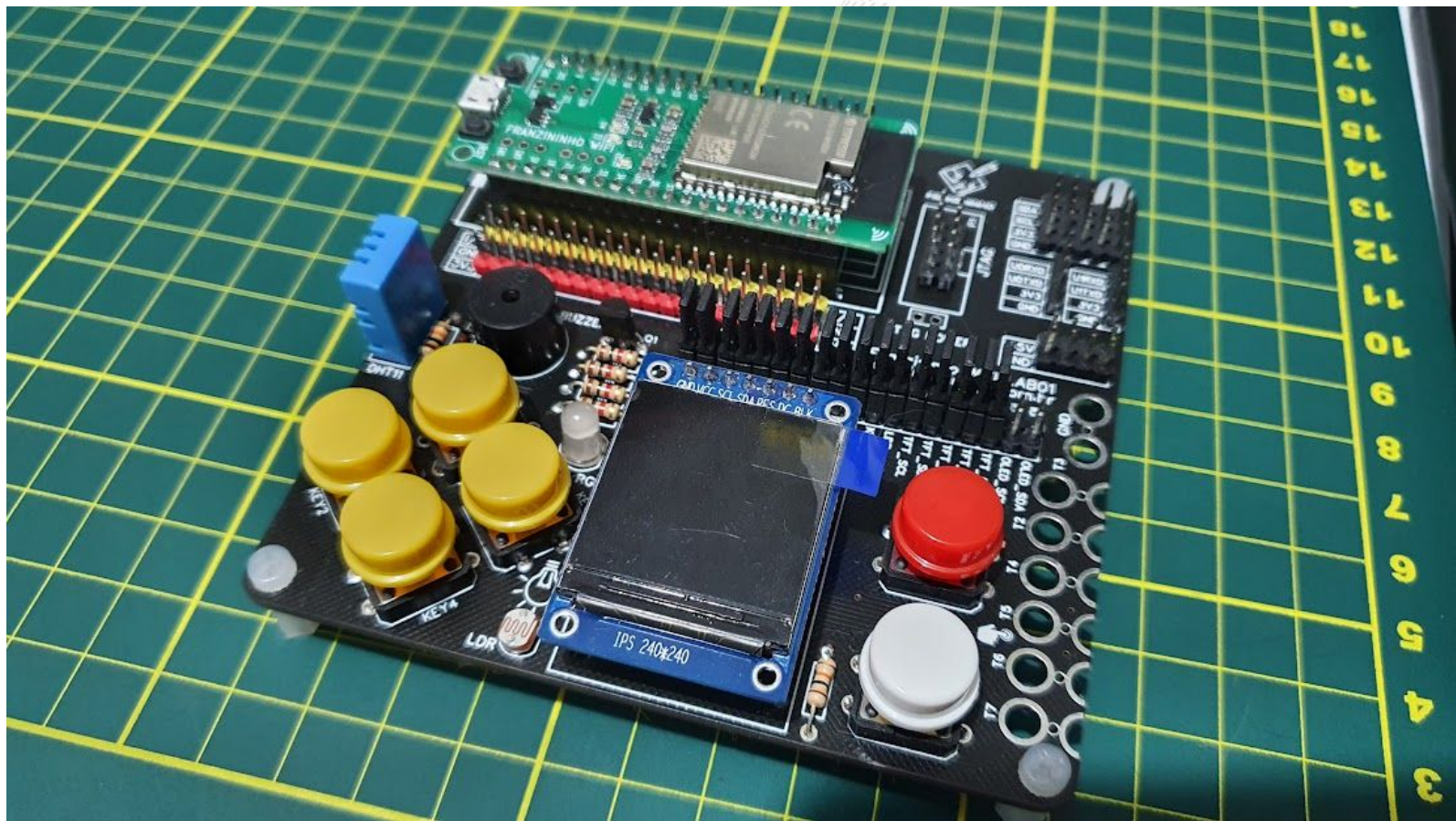
Mais detalhes em: <https://docs.franzininho.com.br>

Circuito



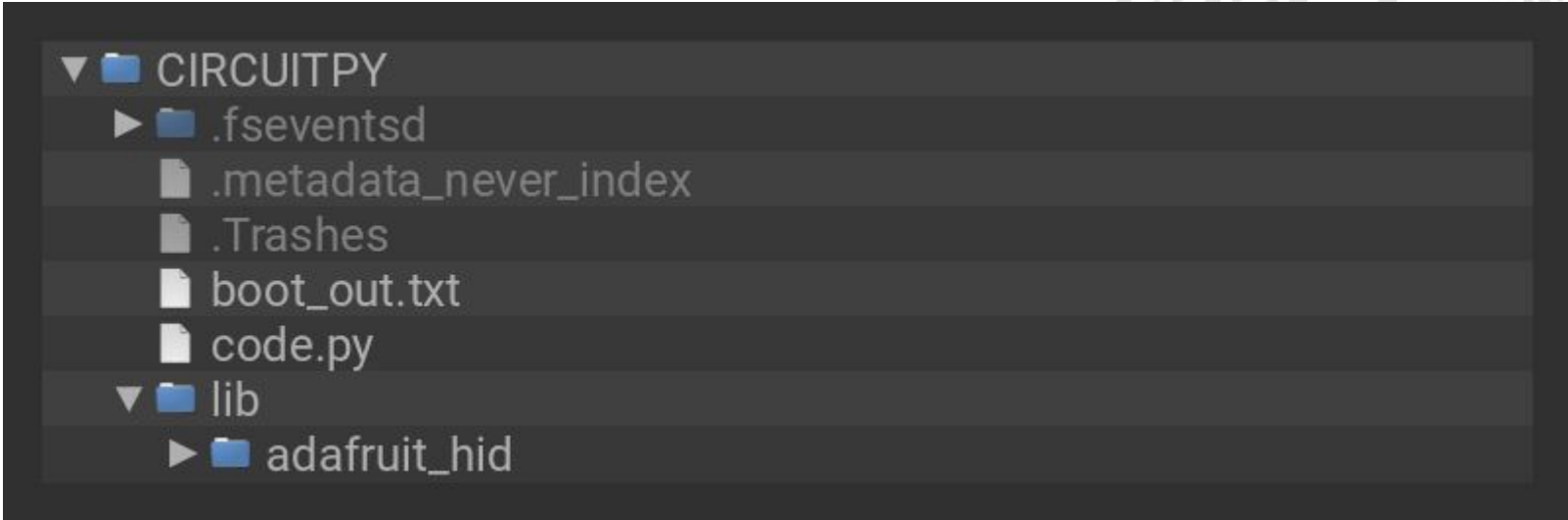
fritzing

Franzininho LAB 01



Biblioteca

- Adafruit HID



```
▼ CIRCUITPY
  ► .fseventsd
  .metadata_never_index
  .Trashes
  boot_out.txt
  code.py
  ▼ lib
    ► adafruit_hid
```

<https://learn.adafruit.com/circuitpython-essentials/circuitpython-hid-keyboard-and-mouse>

<https://circuitpython.org/libraries>

Exemplo 1

- Teclas de atalho

```
while True:
    if key1.value == False:
        kbd.send(Keycode.WINDOWS, Keycode.TAB)

    if key2.value == False:
        kbd.send(Keycode.CONTROL, Keycode.C)

    if key3.value == False:
        kbd.send(Keycode.CONTROL, Keycode.V)

    if key4.value == False:
        kbd.send(Keycode.RIGHT_ARROW)

    time.sleep(0.1)
```

Exemplo 2

- ConsumerControl

```
while True:
    if key1.value == False:
        cc.send(ConsumerControlCode.VOLUME_INCREMENT)
        time.sleep(0.1)

    if key2.value == False:
        cc.send(ConsumerControlCode.BRIGHTNESS_DECREMENT)
        time.sleep(0.1)

    if key3.value == False:
        cc.send(ConsumerControlCode.BRIGHTNESS_INCREMENT)
        time.sleep(0.1)

    if key4.value == False:
        cc.send(ConsumerControlCode.VOLUME_DECREMENT )
        time.sleep(0.1)

    if key5.value == False:
        cc.send(ConsumerControlCode.MUTE)
        time.sleep(0.1)

    if key6.value == False:
        cc.send(ConsumerControlCode.PLAY_PAUSE)
        time.sleep(0.1)

    time.sleep(0.1)
```


Exemplo de teclados



INFINITY MACRPAD 1.1



CircuitPython

- O CircuitPython foi projetado tendo em mente a educação;
- Facilidade na edição e manutenção de códigos;
- Serial Console + REPL;
- Armazenamento interno para registro de dados, reprodução de áudio e outras formas de interação com arquivos;
- Mais de 280 bibliotecas;
- Tradução para diversas línguas

Referências

- <https://docs.circuitpython.org/projects/hid/en/latest/index.html>
- <https://learn.adafruit.com/customizing-usb-devices-in-circuitpython/hid-devices>
- <https://learn.adafruit.com/circuitpython-essentials/circuitpython-hid-keyboard-and-mouse>
- https://github.com/adafruit/Adafruit_CircuitPython_HID

Dúvidas?

Contato:

Instagram: fabiosouza.io